

ETKİLEŞİMLİ IŞIN İZLEME

Ömer ÇAKIR¹

Cemal KÖSE²

Bilgisayar Mühendisliği Bölümü
Mühendislik-Mimarlık Fakültesi
Karadeniz Teknik Üniversitesi, 61080, Trabzon

¹e-posta: cakiro@ktu.edu.tr

²e-posta: ckose@ktu.edu.tr

Anahtar sözcükler: Etkileşimli Işın İzleme, Paralel Işın İzleme, Gölgeleme, Doku Kaplama.

ÖZET

Bu bildiriye, bir yerel ağdaki kişisel bilgisayarlar kullanılarak gerçekleştirilen, etkileşimli ışın izleme uygulaması sunulmaktadır. Etkileşimli ışın izleme uygulanan yöntem, ekrandaki her bir pikselden geçen ışınları nesne uzayında izlerken, sistem kaynaklarından etkin biçimde yararlanmayı hedeflemektedir. Gerçeklenen etkileşimli sistemin tasarımı, kullanılan modelleme, aydınlatma, gölgeleme, dokulandırma ve görüntüleme teknikleriyle birlikte verilmektedir. Burada ışın izleme yönteminin hızlandırılması için iki temel hızlandırma yönteminden yararlanılır. Bunlardan ilki ışın izleme yönteminin hızlandırılması veya ışın-nesne kesişim testlerinin sayısının azaltılmasıdır. İkinci hızlandırma yöntemi ışın izleme yönteminin paralelleştirilmesidir. Bu çalışmada etkileşimli ışın izleme için her iki hızlandırma yönteminden de yararlanılmıştır.

1. GİRİŞ

Bilgisayar grafiklerinde üç boyutlu (3B) görüntü üretim yöntemlerinden biri olan Işın İzleme (Ray Tracing), ilk olarak 1968 yılında Appel tarafından ortaya atılmıştır [1]. Appel'in ışın izleme yöntemi 3B nesnelere için yüzey boyama ve gölgeleme yapabilmesine rağmen, ışının aynasal yansıma ve kırılma özelliklerini modelleyemiyordu. Whitted ve Kay'ın çalışmaları sonucunda, ışın izleme algoritması ışının aynasal yansıma ve kırılma özelliklerini de modelleyebilecek şekilde geliştirilmiştir. Işın izleme yöntemi bu özellikleriyle, 3B görüntü üretimi teknikleri içinde gölge belirleme, aynasal yansıma, kırılma ve katı cisim üretimi için en etkin yöntemlerden biridir.

Etkileşimli görüntü üretimi için tek işlemci yeterli olmadığından, yapılan çalışmalarda paralel bilgisayarlar kullanılmıştır. Işın izleme yöntemi herhangi bir pikselin renk bilgisini diğerinden bağımsız hesapladığından paralel hesaplama için oldukça uygun bir yöntemdir. Sistemin performansı ilgilenilen görüntünün karmaşıklığı ve görüntü boyuna bağlı olarak değişmesine rağmen seçilen paralel sistemin ölçeğine göre oldukça olumlu sonuçlar elde edilmiştir.

Etkileşimli ışın izleme ile ilgili yapılan çalışmaların genellikle paralel çalışan geleneksel çok-işlemcili sistemler üzerinde olduğu görülmektedir. Bu sistemler paralel hesaplamının etkinliği açısından daha başarılı olsa da kısa zamanda demode olmakta ve kolayca güncelleştirilememektedir [2]. Oysa günümüzdeki kişisel bilgisayarlardaki işlemciler geleneksel paralel bilgisayar sistemlerinde kullanılan işlemcilerle göre oldukça güçlüdür. Dolayısıyla geleneksel paralel işlemciler yerine, kişisel bilgisayarlardan oluşan paralel sistemleri kullanmak daha avantajlı olmaktadır. Böylece bilgisayarlar yavaş kaldığında sisteme kolayca yüksek hızlı yeni bilgisayarlar eklenebilmektedir. İstenirse tüm sistem kolayca güncellenebilmekte ve geliştirilmiş yazılımlar sistemde çoğu zaman güncelleme gereksizdir kullanılabilmektedir. Geleneksel paralel işlemcilerde ise sistemin değiştirilmesi ile birlikte çoğu zaman geliştirilen yazılımların da büyük ölçüde değişmesi gerekmektedir. Bu çalışmada etkileşimli bilgisayar grafikleri için en uygun hesaplama ortamını sunan, güncel ağ teknolojisi kullanılarak birleştirilen kişisel bilgisayarlardan oluşan, paralel bilgisayar sistemi kullanılmıştır.

2. IŞIN İZLEME

Işın izleme yöntemi, 3B görüntüler üretmek için, bakış noktasından görüntü düzlemindeki her bir pikselden geçecek şekilde nesne uzayına ışınlar yollar. Bu ışınlara "birincil ışınlar" denir. Üretilen görüntüdeki herhangi bir pikselin rengi, o pikselden geçen birincil ışın ile görüntüsü üretilecek nesnelere ait üçgenler arasında kesişim testleri yapılarak görünmeyen yüzeyler kaldırılarak belirlenen, bakış noktasına en yakın üçgenin rengine, eğer bu üçgen ışını yansıtıcı/kırıcı özelliğe sahipse ışının bu üçgenden yansıyıp/kırılıp çarptığı ilk üçgenin rengine, bakış noktasının ve ışık kaynağının konumuna göre belirlenir. Burada yüzeyin yansıtıcılığına ve geçirgenliğine bağlı olarak ikincil, üçüncül ve diğer ışınlar tam görüntünün üretimi için birincil ışındakine benzer şekilde dikkate alınmaktadır. Bu çalışmada gerçekçi görüntülerin üretimi için kullanılan etkileşimli ışın izleme yöntemi yansıma, gölge testi, doku kaplama, hızlandırma yöntemlerini içermektedir.

2.1. YANSIMA MODELİ

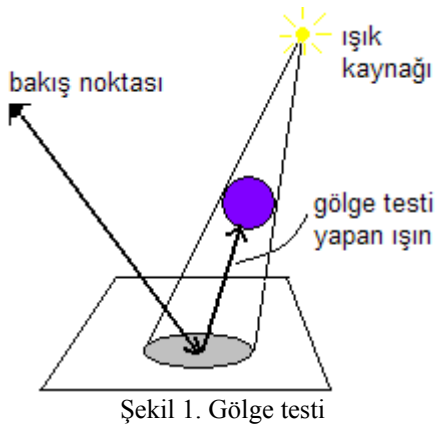
Işın izleme yansımaya ile görünen nesnelere modelleme yeteneğine sahip bir yöntemdir. Işın izleme ile aynasal yansımaya modellemek için yansıyan ışının doğrultusu hesaplanır. R_i doğrultulu gelen ışının N normaline sahip yüzeyden R_r yansımaya doğrultusu

$$R_r = R_i - 2(R_i \cdot N)N \quad (1)$$

ile hesaplanır. Işın yansıtıcı özelliğe sahip bir yüzeye çarptığında (1) bağıntısı ile yansımaya doğrultusu hesaplanır. Işının yeni başlangıç noktası yüzey üzerindeki kesişim noktasından, yeni doğrultusu yansımaya doğrultusu boyunca hareketi sonucu çarptığı ilk yüzeyin görüntüsü yansıdığı yüzeyde görülür.

2.2. GÖLGE TESTİ

3B uzaydaki nesnelere üçgen poligonlardan oluştuğu varsayılırsa, ışın izlemede ışının üçgenlerle oluşan kesişim noktasının başka bir nesneye ait üçgenin gölgesinde kalıp kalmadığı belirlenirken, Şekil-1 de görüldüğü gibi kesişim noktasından ışık kaynağına gölge testi yapan ışınlar yollar. Yollanan ışın nesnelere ait herhangi bir üçgenle kesişiyorsa yüzey bu nesnenin gölgesinde kalıyor demektir. Işık kaynağı noktasal ise gölgeler keskin olur. Noktasal olmayan, poligon ışık kaynakları kullanıldığında ise daha fazla test ışını ile gölgeler yumuşatılmaktadır.



Şekil 1. Gölge testi

2.3. DOKU KAPLAMA

Nesnelerin dokulandırılması üretilen görüntülerin gerçekçiliğini daha da arttırmaktadır. Nesnelerin doku kaplanması için barisentrik koordinatlardan yararlanılmıştır. Işın-üçgen kesişim testi olarak kullanılan algoritma sadece ışının üçgene t uzaklığını değil aynı zamanda (u,v) barisentrik koordinatları da hesaplar [3]. Daha sonra hesaplanan bu koordinatlar kullanılarak doku kaplama yapılmıştır [3, 4].

2.4. HIZLANDIRMA YÖNTEMLERİ

Işın izleme yöntemi ile görüntü üretimini ve etkileşimli ışın izlemeyi hızlandırmak için bazı yöntemler kullanılmıştır. Genellikle bu yöntemler ışın nesne kesişim test sayısını azaltarak hızlandırma yapmaktadır. Bu hızlandırma yöntemlerinden biri

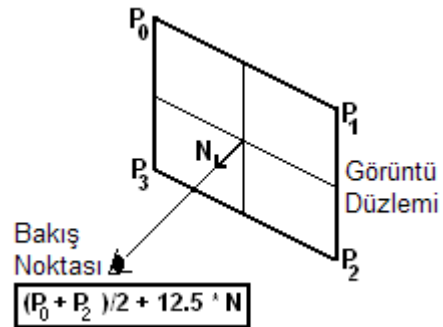
Çevreleyen Hacim (Bounding Volume) yöntemidir. Bu yöntemde göre ışınlar ile nesnelere oluşturan üçgenler arasında kesişim testi yapmadan önce ışınların nesneyi çevreleyen küre, küp gibi hacimsel bir şekil ile kesişip kesişmediğine bakılır. Etkileşimli ışın izleme uygulamasında kullanılan ortamdaki nesnelere için çevreleyen hacim olarak dikdörtgen prizma kullanılmış ve önce ışın-dikdörtgen prizma kesişim testi yapılmıştır [5]. Işın, dikdörtgen prizma ile kesişmiyorsa nesneyi oluşturan üçgenler için kesişim testi yapılmamış ve böylece kesişim testleri hızlandırılmıştır.

Işın izleme yönteminde zamanın çoğu ışın-nesne kesişim testlerine harcanmaktadır. Üretilen görüntülerdeki nesnelere üçgenlerden oluştuğu için hızlı ışın-üçgen kesişim testi olan Möller'in yöntemi kullanılmıştır. Möller'in yönteminde kesişim testi, kesişim noktası için barisentrik koordinatlar ve ışın ifadesi arasındaki $t(u,v) = o + td$ eşitliğinin lineer cebirsel çözümüne dayanır. En son ışının üçgene t uzaklığı ve (u,v) barisentrik koordinatlar için vektörel ifadeler elde edilir [3].

Etkileşimli ışın izlemedeki transformasyonlar için yalnızca bakış noktası ve görüntü düzlemine ait dört köşe noktasının yeni koordinatları hesaplanmaktadır. Böylece 3B nesnelere koordinatları değişmemekte ve bu durum etkileşimli ışın izlemeyi oldukça hızlandırmaktadır.

3. ETKİLEŞİMLİ IŞIN İZLEME

Etkileşimli ışın izleme, belli bir bakış noktasındaki gözlemcinin klavye/mouse kullanarak belli dönme, ilerleme hareket komutlarıyla gözlemlediği ortamla arasındaki etkileşim olarak tanımlanabilir.



Şekil 2. Yeni görüntü düzlemi ve bakış noktası

3.1. GÖRÜNTÜLEME MODELİ

Etkileşimli ışın izlemede bakış noktası ve görüntü düzleminin yeni koordinatlarının doğru olarak hesaplanması büyük önem taşımaktadır. Görüntü düzlemi, Şekil-2 de görüldüğü gibi kendisine ait dört köşe noktası kullanılarak ifade edilmiştir. Yani ilerleme veya dönme hareketleri yapılırken görüntü düzleminin yeni koordinatlarını belirlemede bu köşe noktalarının koordinatlarını kullanmak yeterli olmuştur. Köşe noktaları sol üst köşeden başlanarak

P_0 , P_1 , P_2 ve P_3 ile temsil edilmiştir. Görüntü düzlemi 8×6 birim boyutlarında; bakış noktası görüntü düzleminin merkezine dik ve insan gözünün görüş alanı 35° için min. uzaklık olan 12.5 birim uzaklıkta alınmıştır [6].

Kullanıcı arayüz birimi oluşturulmuş ve etkileşimli ışın izleme için kullanıcının bu arayüz üzerinden belli tuşları kullanarak, cisim uzayında istenilen eksen etrafında dönmesi ve ileri-geri ilerlemesi sağlanmıştır.

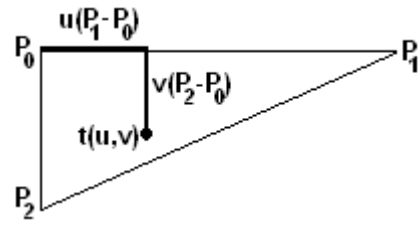
3.2. ETKİLEŞİMLİ GÖRÜNTÜLEME

Kullanılan görüntüleme modelinde dönme hareketinin bakış noktası etrafında olacağı varsayıldığından öncelikle P_0 , P_1 , P_2 , P_3 ten bakış noktası çıkarılır. Elde edilen koordinatlar dönme işlemi için gereken rotasyon matrisi ile çarpılıp bakış noktası tekrar eklenir. Böylece dönme hareketi sonucu görüntü düzleminin yeni P_0 , P_1 , P_2 , P_3 noktaları elde edilir.

Bakış noktası her zaman görüntü düzleminin merkezine işaret ettiğinden yani görüntü düzleminin göre belirlendiğinden görüntü düzleminin koordinatları değişince bakış noktasının koordinatları da değişir. Bakış noktasının yeni koordinatlarını bulmak için öncelikle yeni görüntü düzleminin merkezinin koordinatları belirlenmelidir. Görüntü düzleminin merkezi $(P_0+P_2)/2$ veya $(P_1+P_3)/2$ ile bulunur. Bakış noktası görüntü düzleminin merkezine dik konumda olduğu için görüntü düzleminin yeni normali bulunmalıdır. Çünkü dönme hareketi sonucu normal de değişmiştir. Yeni normal $(P_1-P_0) \times (P_2-P_1)$ vektörel çarpımı ile hesaplanabilir. Normal normalize edilerek boyu 1 birim yapılır. Normal bakış noktasının görüntü düzleminin uzaklığı olan 12.5 ile çarpılıp görüntü düzleminin merkezine eklendiğinde bakış noktasının yeni koordinatları bulunur.

İlerleme hareketinde de hem görüntü düzleminin hem de bakış noktasının koordinatları değişmektedir. İlerleme hareketinde görüntü düzleminin yeni P_0 , P_1 , P_2 , P_3 noktaları ve bakış noktasının koordinatları ilerleme doğrultusuna göre belirlenir. İlerleme doğrultusu ise geri yönde ilerlemede görüntü düzleminin normali doğrultusunda; ileri yönde ilerlemede ise normalin tersi doğrultuda olur. İleri yönde k birim kadar ilerlenecekse hesaplanan normal k ile çarpılıp P_0 , P_1 , P_2 , P_3 noktalarından çıkarılır (geri yön için toplanır). Ardından dönme hareketinde olduğu gibi bakış noktasının yeni koordinatları hesaplanır.

Etkileşimli ışın izlemede görüntü düzlemi ve bakış noktasının koordinatları değiştikçe bakış noktasından yollanan birincil ışınların doğrultusu da değişmektedir. Herhangi bir birincil ışının doğrultusu hesaplanırken ışının geçtiği pikselden bakış noktası çıkarılıp normalize edilir. Görülüyor ki birincil ışınların doğrultusu hem görüntü düzleminin hem bakış noktasına bağlıdır.



Şekil 3. Barisentrik koordinatlar

İlerleme/dönme hareketinde yeni bakış noktasının nasıl hesaplandığı yukarıda anlatılmıştı. Piksel koordinatları hesaplanırken P_0 , P_1 , P_2 , P_3 noktaları kullanılır ve barisentrik koordinatlar kavramından faydalanılır. Şekil-3 te görüldüğü gibi üçgenin herhangi bir köşesinden başlanıp kenarları belli katsayılarla çarpılarak istenilen bir noktaya gidilebilir. Barisentrik koordinatlar (2) ile hesaplanır [5]. Burada $u \geq 0$, $v \geq 0$, $u+v \leq 1$ şartı sağlanmalıdır.

$$t(u,v) = P_0 + u(P_1-P_0) + v(P_2-P_0) \quad (2)$$

$$P_0 + ((P_1-P_0)/640)*x + ((P_2-P_0)/480)*y \quad (3)$$

Bu bilgiler ışığında, örneğin üretilecek görüntü 640×480 çözünürlükte olduğunda piksel koordinatları, görüntü düzleminde P_0 noktasına pikselin (x,y) değerlerine bağlı olarak x eksenini boyunca $((P_1-P_0)/640)*x$; y eksenini boyunca da $((P_2-P_0)/480)*y$ değeri eklenerek (3) ile hesaplanır.

4. PARALEL IŞIN İZLEME

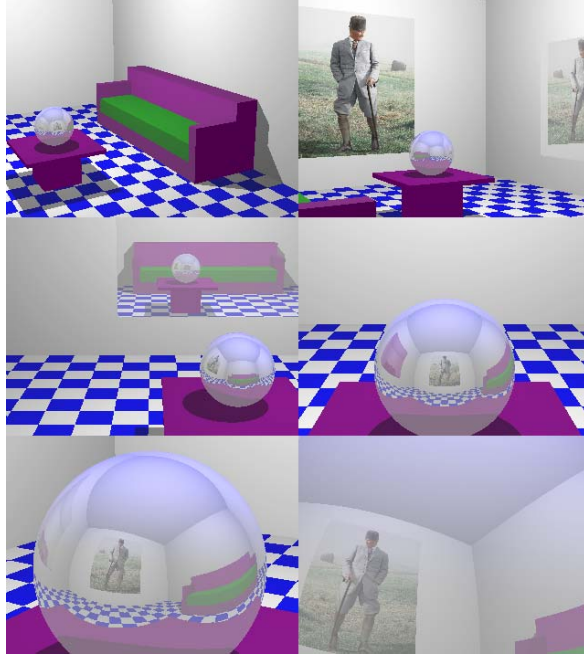
Paralel bilgisayarlar ışın izleme yönteminin daha da hızlandırılması için kullanılmaktadır. Yapılan çalışmalarda etkileşimli ışın izlemenin gerçek zamanlı olabilmesi için tek bilgisayarın yetersiz olduğu görülmüştür. Bir yerel ağ üzerindeki bilgisayarlar paralel koşturularak gerçek zamanda ışın izleme elde etmek amaçlanmıştır.



Şekil 4. Master-slave yaklaşımı

Işın izlemede herhangi bir pikselin renk bilgisi diğerinden bağımsız hesaplandığından piksel mertebesinde paralel hesaplama yapmak mümkündür. Fakat hesaplama karmaşıklığı pikselden piksele değişmektedir. Dolayısıyla etkileşimli izleme için dinamik yük dengelemesi sağlayan Master-Slave yaklaşımı kullanılmıştır. Şekil-4 te görüldüğü gibi Master süreç problemi alt parçalara ayırır ve her bir Slave sürece hesaplamaya başlaması için ihtiyaç duyduğu başlangıç verilerini yollar. Slave'ler hesaplamayı bitirince sonuçları Master sürece yollar ve o da sonuçları bir araya getirir [7].

Paralel programlama için iki program geliştirilmiştir. Ana bilgisayarda koşan Master program, kullanıcı herhangi bir hareket tuşuna bastığında ağdaki bilgisayar sayısına bağlı olarak üretilecek görüntüyü parçalara ayırır ve her bir Slave bilgisayara hangi pikselleri hesaplayacağını, hangi hareketi yerine getireceğini bildirir. Ağdaki bilgisayarlar kendi işlerini tamamlayıp ana bilgisayara sonuçları yollarlar ve ana bilgisayar da gelen sonuçları bir araya getirerek görüntüyü üretir. Kullanıcının her tuşa basışında bu işlemler tekrarlanır.



Şekil 5. Etkileşimli ışın izleme ile üretilen görüntüler

5. UYGULAMALAR

Etkileşimli ışın izleme uygulaması olarak içinde basit nesnelere ve aynasal özelliğe sahip nesnelere olan bir sanal ortamda kullanıcının gezinmesi yapılmıştır. Gerçek zamanlı etkileşimli bir uygulama amaçlandığı için, ışın izleme yapılan ortamda mümkün olduğunca çok karmaşık nesnelere bulunmamaktadır. Görüntüler önce tek bilgisayar kullanılarak, ardından yerel ağdaki bilgisayarlar kullanılarak üretilmiştir. Şekil-5 te etkileşimli ışın izleme ile üretilmiş 320x240 çözünürlükte 6 görüntü görülmektedir. Görüntülere dikkat edilirse doku kaplama, aynasal yansıma ve gölgeleme gibi özellikleri olduğu görülür.

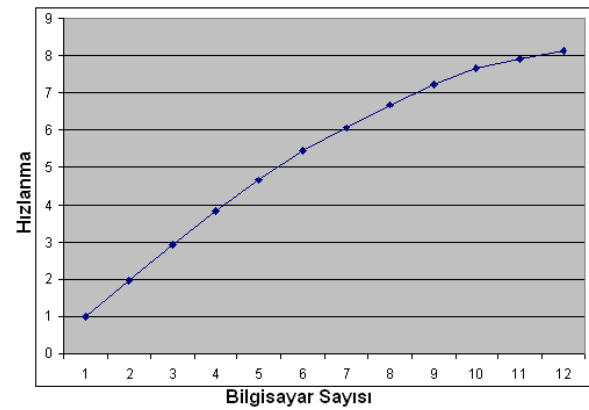
6. SONUÇLAR ve ÖNERİLER

Etkileşimli ışın izleme uygulaması ilk olarak Pentium-IV 2.4 GHz işlemcisi 256 MB ana belleği olan bilgisayar üzerinde test edilmiş ve hızlandırma yöntemleri kullanılmıştır. Görüntüler 1,2,3,4,5,6, 8, ve 12 bilgisayar kullanılarak üretilmiş ve hesaplama süreleri ve hızlanma (speedup) değerleri Tablo-1 de verilmiştir. Etkileşimli görüntü üretimi için, Şekil-6 da görüldüğü gibi, bilgisayar sayısının artmasına bağlı olarak oldukça olumlu hızlanmalar sağlanmıştır.

Ayrıca Tablo-1 de saniyede üretilen görüntü "FPS (frame per second)" sayıları verilmiştir. FPS değerlerine bakıldığında bilgisayar sayısı 6 olduğunda 2 fps, 12 olduğunda da 3 fps ye ulaşıldığı görülür.

Tablo 1. Hızlanma ve FPS değerleri

	Bilgisayar Sayısı							
	1	2	3	4	5	6	8	12
Süre	2.61	1.33	0.89	0.68	0.56	0.48	0.39	0.32
FPS	0.38	0.75	1.12	1.47	1.79	2.08	2.56	3.13
Hızlanma	1.00	1.96	2.93	3.84	4.66	5.44	6.69	8.16



Şekil 6. Hızlanma (speedup) grafiği.

Hedeflenen performansın biraz altında kalınmasının önemli nedeni kullanılan ağın band genişliğinin sınırlı olmasıdır. Bu durumda, daha küçük boyutlu görüntüler üretilerek, saniyede üretilen görüntü kolayca 15-20 civarına çıkmaktadır. Üretilen görüntülerin daha gerçekçi olması için poligon sayısının artırılması, gölge yumuşatma, doku kaplamada karşılaşılan aliasing probleminin giderilmesi gibi bazı yöntemlerden yararlanılabilir.

7. KAYNAKLAR

- [1] Glassner A., An Introduction To Ray Tracing, Ninth printing, Morgan Kaufmann Publishers, San Francisco, 2002.
- [2] Parker S, Parker M, Livnat Y., Sloan P, Hansen C., Shirley P., "Interactive Ray Tracing for Volume Visualization". IEEE Trans. Vis. Com. Graph. 5(3): 238-250, 1999.
- [3] Möller T., Trumbore B., "Fast Minimum Storage Ray-Triangle Intersection", Journal of Graphics Tools, 2(1), 21-28, 1997.
- [4] Çakır Ö., "Yerel Ağdaki Kişisel Bilgisayarlarla Paralel Işın İzleme", KTÜ, 2004.
- [5] Möller T., Haines E., Real-Time Rendering, Second Edition, A. K. Peters, 2002.
- [6] Ankrum D. R., "Viewing Distance at Computer Workstations", WorkPlace Ergonomics, 10-12, Sept/Oct 1996.
- [7] Wilkinson B., Allen M., Parallel Programming, Prentice Hall, New Jersey, 1999.